A Newly Telesecurity of VoIP using SIP protocol in VPN

Sung-Ki Lee*, Doh-Yeun Hwang*, Seung-Ryong Yi*,
Seung-Sun Yu**, and Hoon-sung Kwak*

* Department of Computer Engineering, Chonbuk University, Jeonju, Korea
(Tel : +82-63-270-2417; E-mail: www2www@empal.com)
** Department of Image Engineering, Chonbuk University, Jeonju, Korea
(Tel : +82-63-270-2411; E-mail: yss2590@hanmir.com)

Abstract: The VoIP (Voice over IP) is being used world-widely and already put to practical use in many fields. However, it is needed to ensure the security of VoIP call in special situations. It is relatively difficult to eavesdrop commonly used PSTN network in that a 1:1 circuit connects it. However, it is difficult to ensure the security of a call on Internet because many users are connected to the Internet concurrently. This paper suggests a new model for Internet telephony to prevent eavesdrops, using VoIP (using SIP protocol) with the use the VPN protocol and establish the feasibility of its practical use comparing it with the conventional Internet telephony.

Keywords: VoIP, Telesecurity, SIP, VPN

1. INTRODUCTION

In late 1980s, General public started to become aware of the existence of the Internet, which became the network that links different countries around the globe. The Internet is becoming more important in our day-to-day life, and its related industries and technology are also advancing rapidly. Among these technologies, the use of Internet Telephony (VoIP) is growing exponentially, threatening the conventional PSTN network. Internet telephony has the advantages of cheaper calls than PSTN, cheaper non-voiced services that are provided on the Internet and will become important in our daily life as the next generation communicational technology. However, Internet telephony (VoIP) is provided through links of shared network and is open to security risks. In this paper we have created Internet telephony terminals that use Virtual Private Network (VPN) to eliminate these risks. PPTP (Point-to-Point Protocol) is used on SIP protocol stack enabled Internet telephony terminal to test its performance and feasibility.

2. VPN ENABLED VOIP TERMINAL HARDWARE

To realize the VPN enabled VoIP terminal, its hardware component is designed as in Fig. 1. As shown in Fig. 1, the hardware consists of a main-board and a sub-board. The sub-board consists of a process module and the main board has an Audio DSP module, an Ethernet module, a SLAC/SLIC module, a power module, 2 connections for network and 2 analog I/O connections for the telephone terminal. Detailed module design is as the following; in Fig. 2 detailed processor module design is shown. In the processor module, MPC850 from Motorola rated at 50 Mhz is used as the main processor. 2 MB of FROM and 8 MB of SDRAM is used for memory. Also Ethernet ports are connected to mpc850, one for the Internet and one for local LAN.

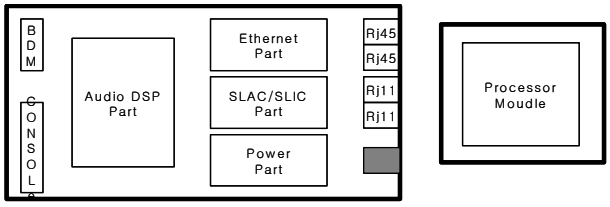

Fig. 1 Logical Hardware Design Block Diagram

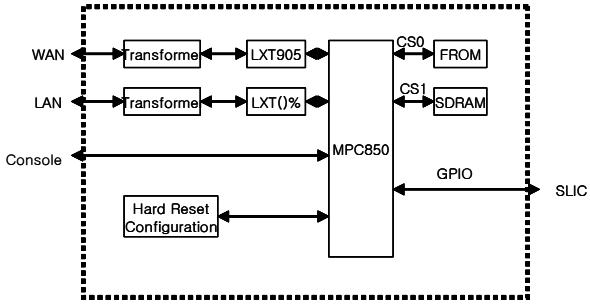

Fig. 2 Detailed Processor Module Design

Detailed design for the Audio DSP module of the main-board, which controls audio packet processor, is shown in Fig. 3. AC4830x-C from AudioCodes is used as the Audio packet processor. This processor is connected to an external memory module (128Kbytes SRAM, CY7C10 21V3-12Z) and uses an external clock rated at 16.384 Mhz.

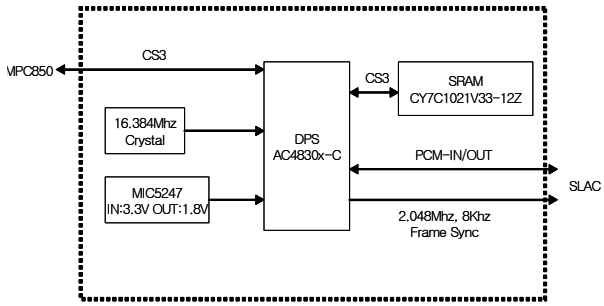

Fig. 3 Detailed Audio Module Design

To convert digital audio signal to analog audio and vice versa, SLAC/RSLIC module is designed as shown in Fig. 4 using a SLAC (model MC14LC5480) from Motorola and a RSLIC (model HC55185) from Intel. SLAC receives analog audio signals from RSLIC, converting the signal to digital and sending them to the audio packet process (AC4830x-C), it also converts digital signals from the audio packet process to analog and sends them to RSLIC. Shown in Fig. 5 is a real-life 2-layered design of the VoIP terminal.

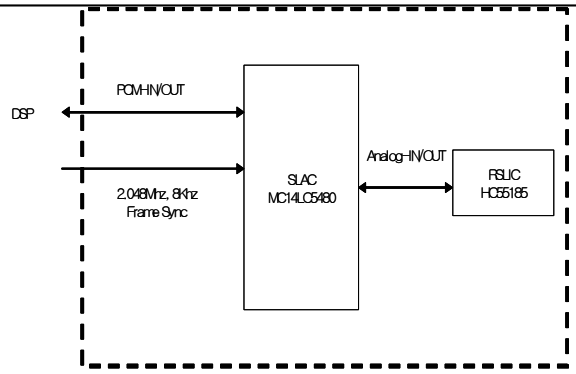

Fig. 4 Detailed Design of the SLAC/RSLIC module

Fig. 5 VPN Enabled VoIP Hardware

3. PROTOCOL STACK OF THE VPN ENABLED VOIP TERMINAL

3.1 PPP (Point-to-Point Protocol)

PPP is capable of transferring Multi-protocol datagram and provides encapsulation, it consists of LCP (Link Control Protocol) which controls PPP connections and flow, and NCP (Network Control Protocol) which negotiates with the network layer. To enable different protocols in the PPP frame, PPP Encapsulation has its frame structure as shown in Fig. 6. It consists of a protocol field, an information field, and a padding field. The protocol field is used to distinguish data (1~2 octet), and informs the protocol used in the information field. Information field has data for the protocol specified in the protocol field and has MRU (Maximum Receive Unit) with the default value of 1500ocetes. This value can be changed by LCP (Link Control Protocol). Padding field's padding method is determined by different protocols.

LCP is the protocol responsible for PPP link configuration, maintenance and termination, and its state transition diagram is shown in Fig. 7. The Dead state is before or after a connection, and the process proceeds to the next state once the physical layer ready. The Establish stage configures the link by exchanging configuration packet between both ends. The negotiated Configuration Option value will be determined the authentication protocol or the absence of one. At the Authentication stage, the client authenticates itself to the server before transferring the packet to the network layers. The protocol to use is determined in the previous stage by exchanging LCP packets, and if authentication fails, process proceeds to the termination stage.

At the Network stage, the network layer protocol, determined by exchanging NCP packets, is set and corresponding packet configuration information is fetched so that transfer can start. At the Termination stage, if authentication fails or termination is chosen in NCP, PPP link is terminated by exchanging termination packets. NCP is a network protocol specified in Fig. 7, it is used to configure the network layer protocol. In this study, IP protocol is used, and IPCP is used to retrieve information from the server and configures IP layer. Retrieved information is usually client's IP address, network mask, gateway address etc. Once IPCP stage of NCP is successful, the protocol field value is set to 0x0021 as shown in Fig. 6.

Fig. 6 PPP Frame Structure

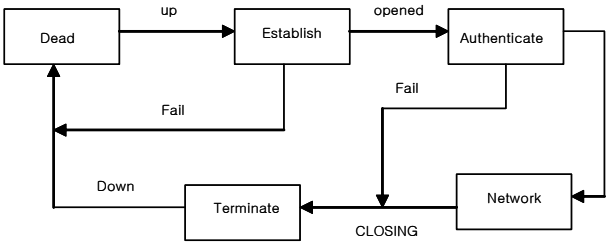

Fig. 7 LCP State Chart Diagram

3.2 PPTP

PPPT is one of VPN (Virtual Private Network) protocol that allows the transfer PPP frame that is encapsulated as IP datagram. This protocol uses TCP data called control connection messages, which is responsible for tunnel creation, maintenance, and termination. PPTP can be divided into Control connection that controls PNS (PPTP Network Server) and PAC (PPTP Access Concentrator), and Tunneling between PNS and PAC. To control PPTP, Control connection massage is sent before tunneling between PAC and PNS is established, it's responsible for tunnel creation, maintenance, and termination over the TCP session. Destination port 1723 is used and can be initiated from either PNS or PAS. Tunneling allow users at both ends to send/receive PPP frame to PNS over the Internet between PAC and PNS as if they are on a leased private line. The PPP frame is encapsulated in GRE (enhanced GRE header) and transmitted, with the IP header attached, between PAC-PNS through Ethernet. The PPTP client is creating different PPTP frames according to the different types of PPTP servers.

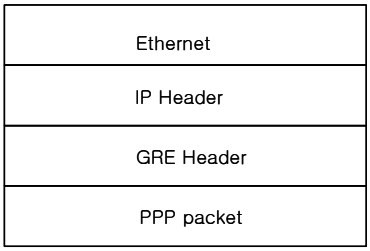

Fig. 8 PPTP Frame Creation Process

3.3 SIP (Session Initiation Protocol) Stack

SIP is used to configure, modify, and terminate media session of VoIP like H.323. However, to use the full functionality of VoIP, it cannot be used alone, but must be combined with other protocols. Most generally used protocol stack is shown in Fig. 9. SIP protocol can be roughly divided into 4 functions and their respective functionality is as the following;

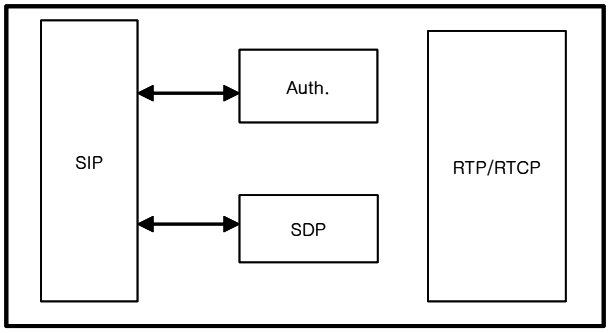

Fig. 9 SIP Stack Structure

Firstly, SIP (Session Initiation Protocol) is responsible for creating, modifying, terminating a multi-media session. SDP (Session Description Protocol) describes a multi-media session, and is included in body part of the SIP message. An Authentication protocol can be used with SIP. When media session is created using these two protocols, RTP (Real-Time Protocol) is used to transfer real-time SDP negotiated format data through path other than the SIP message path. RTP is a real-time data transfer protocol and usually transferred through UDP.

3.4 SIP (Session Initiation Protocol) Stack

VoIP test environment is set as shown in Fig. 10 to test Internet telephony.

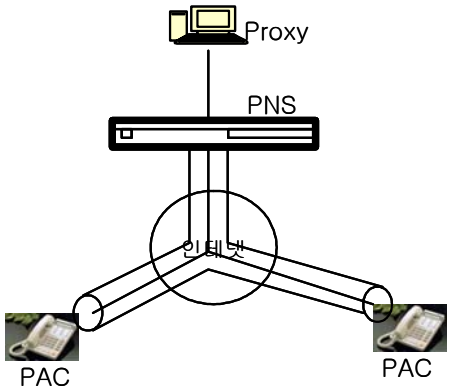

Fig. 10 Test Environment for Internet Telephony

When the system starts, each PAC (PPTP Network Server) creates a control connection with PNS (PPTP Network Server). The Control Connection is used to control PPTP connection before PPTP tunneling is established. To create a Control Connection, firstly a request for a Control connection is sent from PAC to PNS, and on receiving the reply, a request for the configuration of external calls is made. Upon receiving the configuration of external calls, PAC and PNS, using Set-Link-Info message, exchange link information, and finish configuring the control connection. Every VoIP terminal is enabled with PPTP Client (PAC), and SIP protocol. To use SIP protocol to service VoIP, a proxy Server (Registrar functionality included) is needed, and this server is connected to Local LAN where PPTP Server (PNS) resides. When the Control Connection configuration is finished by using PPP (Point-to-Point Protocol) between PAC and PNS, Link Configuration is negotiated, and its supported protocols. Using LCP, configure PPP link, and negotiate whether to use authentication or not. Also the use of compression is negotiated. These negotiated Options during LCP process are used to determine the protocol to use for the next stage. In this study, PNS sends CHAP challenge message to PAC to authenticate since CHAP (Challenge Handshake Authentication Protocol) is used. Then, PAC combines data from PNS and its own user-id and password, and replies with a created CHAP response. If data from received response is correct, sending a CHAP success message terminates the authentication process. During LCP process, it was negotiated that compression is going to be used, so negotiating the compression algorithm is the next process. Negotiation for supported Network Layer protocol also takes place.

In this paper, MPPC is used to compress data, and stateless Mode with 128-bit encryption is used. In Stateless mode, Coherency Count Value of packets in MPPE (Microsoft Point-To-Point Encryption) packet format is different for every packet. To use IP, using IPCP, and the Network layer retrieves network related information from PNS. Information such as private IP of PAC, private network, private gateway IP address, is retrieved and sets PAC network layer. Devices are now ready to connect through VPN. Every PAC can now access the proxy server as in each PAC, private IP address of the proxy server is configured, and once connected to the VPN, private IP and VoIP telephone number of the client is registered in the proxy server. Once the registration is finished, calls can be made by dialing the number through SIP protocol.

During calls between VoIP terminals that are connected to VPN, the voice data is sent/received by the RTP protocol. This voice data travels once caller PAC, using PPTP tunneling, connects to PNS, retrieves private IP address that maps the real destination, and once again PNS using PPTP tunneling, to the called PAC. Data traveling in opposite direction travels in similar fashion. In this study, the protocol stacks of different nodes that the real voice data travels. once receiving voice from caller terminal, digital voice data is retrieved using Audio DSP. Attaching a RTP header to digital voice data, send to the UDP layer. At the UDP layer, a UDP header is attached and sent to the private IP layer. Using VPN IP, a IP header is attached, and sent to the MPPE (encryption) layer. At this encryption layer, encryption of the packet from the private IP layer takes place and a MPPE header is attach. This encrypted packet is sent to the PPP layer. At the PPP layer, a PPP header is attached and sent to the GRE layer. At the GRE layer, the packet is encapsulated in a GRE header and sent to the public IP layer.

At the Public layer, a public IP head is attached using public IP, which is used in the actual Internet. This final IP packet sent to the physical layer, and this Ethernet frame is sent to the Internet network. PNS then receives this Ethernet frame from caller terminal, and processes it in reverse order (Ethernet -> public IP -> GRE -> PPP -> MPPE -> private IP) to retrieve the final destination from the private IP layer. Depending on the destination, Ethernet frame is created in private IP -> MPPE -> PPP -> GRE -> public IP -> Ethernet processes to the called handset. And at the called handset,

Ethernet frame received form PNS, is processed in revered layer order to retrieve the final voice data. Encapsulated IP packets of Ethernet frames, using protocol stack shown in Fig. 11, that are sent/received from th e external Internet network, is structure as shown in Fig. 12. These packets are immune to IP packet hijacking since real data is encrypted.

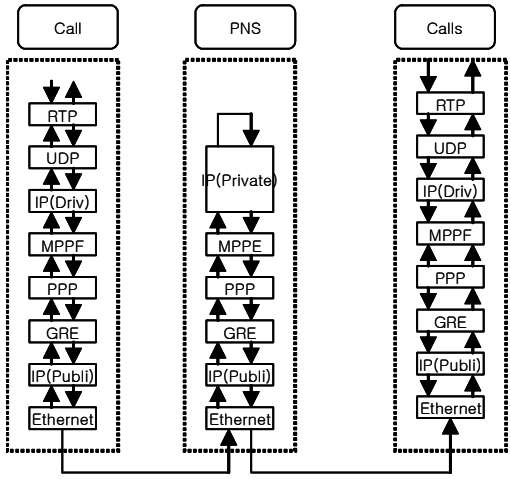

Fig. 11 Protocol Stack for Voice Data Transfer

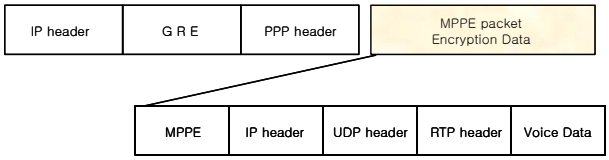

Fig. 12 IP Packet Format of the Encrypted Voiced Data

4. PERFORMACNE EVALUATION

To test the performance of Internet Telephony (VoIP) using VPN (PPTP) as described in this test, its performance was measured using benchmark methods of conventional Internet telephony. Based on given Call patterns, to test the call completion rated in a local LAN setup, a test environment as described in [Figure 16] was setup. DUT (Device Under Test, i.e. VoIP handset) was configured with directcall, DTMF in band signaling, G.723.1 (6.3K) codec settings. And to enable directcall, PNS was enabled at each terminal. As described in HammerIT with call length of 10sec, intercall time of 3sec, start to start time 0sec, creating calls with blast call pattern for 24 hours to test the call completion rate. Every call is connected for 10 seconds, then disconnected for 3 seconds then reconnected for 10 seconds and so on, repeated for 24 hours. In Fig. 13, 'Place Call' means call connection, 'Confirm Path Stimulus' means maintaining the call by sending tones or voice for the testing duration. In this study, call length was set to 10 seconds. If it took less than 10 seconds to send tones and confirm it between DUT1 and DUT2 then it was repeated so that the call length lasted about 15 secondsll figures and tables should be placed after their first mention in the text. Large figures and tables may span across both columns. Scanned images (e.g., line art, photos) can be used if the output resolution is at least 600 dpi.

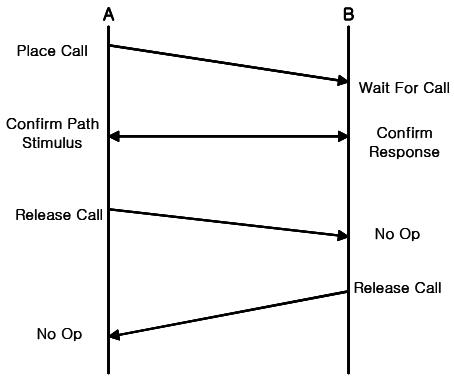

Fig. 13 Test Script for Call Completion Rate from HammerIT

5. CONCLUSION

To test the Eavesdrop-proof Internet telephony using PPTP protocol and SIP protocol of this study, call length was set to 10 seconds if it took less than 10 seconds to send tones from A to B and to confirm them, it was repeated at either A or B so that the whole call length became 15 seconds. This setup was tested for 24 hours, and it was noted that the call was completed 100% of the time. Hence the eavesdropping on Internet telephony, which is its biggest disadvantage, can be overcome with the use of VPN in Internet telephony. However, the PPTP protocol used in this study does not provide encryption for control connection messages, or authentication methods, hence it is exposed to the risk of control connection message hijacking and analysis. It is recommended that the use of IP sec protocol, which provides control connection message encryption and authentication, is more advisable in situations with security risks.

REFERENCES

[1] K. Hamzeh, G. Pall, W. Verthein, J. Taarud, W. Little, G. Zorn, "Point-to-Point Tunneling Protocol (PPTP)", *RFC 2637*, July 1999

[2] W. Simpson, "The Point-to-Point Protocol (PPP)", *RFC 1661*, July 1994

[3] W. Simpson, "PPP LCP Extensions", *RFC 1548*, January 1994

[4] W. Simpson, "PPP Challenge Handshake Authentication Protocol (CHAP)", *RFC 1994*, August 1996

[5] D. Rand, "The PPP Compression Control Protocol (CCP)", *RFC 1962*, June 1996

[6] G. McGregor, "The PPP Internet Protocol Control Protocol (IPCP)", *RFC 1332*, May 1992

[7] G. Pall, G. Zorn, "Microsoft Point-To-Point Encryption (MPPE) Protocol", *RFC 3078*, March 2001